Abstract

Computer systems are typically designed with multiple levels of memory hierarchy. Prefetching has been employed to overcome the latency of fetching data or instructions from or to memory.

5 Prefetching works well for data structures with regular memory access patterns, but less so for data structures such as trees, hash tables, and other structures in which the datum that will be used is not known a priori. The present invention significantly increases the cache hit rates of many important data structure traversals, and thereby the potential throughput of the computer system and application in which it is employed. The invention is applicable to those data structure accesses in which the traversal path is

10 dynamically determined. The invention does this by aggregating traversal requests and then pipelining the traversal of aggregated requests on the data structure. Once enough traversal requests have been accumulated so that most of the memory latency can be hidden by prefetching the accumulated requests, the data structure is traversed by performing software pipelining on some or all of the accumulated requests. As requests are completed and retired from the set of requests that are being traversed, additional

15 accumulated requests are added to that set. This process is repeated until either an upper threshold of processed requests or a lower threshold of residual accumulated requests has been reached. At that point, the traversal results may be processed.